5. TCP Window Scan: Like checking a letter's fine print to see if you're invited!

The `TCP Window` scan refines the technique used in `ACK` scans by analyzing the |`TCP window` size of `RST` packets returned from a target. If a port is open, the `TCP window` size in the `RST` packet is often non-zero, subtly indicating an active listening state, whereas closed ports generally return a zero window size.

This method is useful when more common scans like SYN are blocked, offering an alternative for deducing port status.

Write a bash script that performs a a `TCP Window` scan on a specified test network. The scan should identify potential stealth ports, in the range from `20` to `30`, but exclude ports from `25` to `28`.

- Your script should accept `host` as an arguments `$1`.
- Your script should accept `ports` as an arguments `$2`.
- Your script should accept a range of ports to exclude as an argument `$3`.

Depending on the scanned network, the output could change.

```
┌──(maroua)-[~/0x06_nmap_advanced_port_scans]
└─⚑  ./5-window_scan.sh www.holbertonschool.com 20-30 25-28
[sudo] password for maroua:
Starting Nmap 7.80 ( https://nmap.org ) at 2024-04-19 15:38 CET
Nmap scan report for www.holbertonschool.com (3.233.126.24)
Host is up (0.22s latency).
Other addresses for www.holbertonschool.com (not scanned): 34.234.52.18
52.206.163.162 64:ff9b::22f9:c8fe 64:ff9b::3f23:338e 64:ff9b::3411:7769
rDNS record for 3.233.126.24: ec2-3-233-126-24.compute-1.amazonaws.com

PORT   STATE SERVICE
20/tcp open  ftp-data
21/tcp open  ftp
22/tcp open  ssh
23/tcp open  telnet
24/tcp open  priv-mail
29/tcp open  msg-icp
30/tcp open  unknown

Nmap done: 1 IP address (1 host up) scanned in 0.86 seconds
```

```
#!/bin/bash
sudo nmap -sW -p$2 --exclude-ports $3 $1
```